# Motion Estimation

**Version 2.0.1**

Motion estimation is a very important task in video analysis. It can be used to find the motion fields, to identify moving objects and to find their velocity. The motion displacement vector definition is shown in Figure 1. It has two components *Ux,Uy* along the *x* ,*y* coordinates. The motion velocity is equal to the magnitude of the motion vector $U = \frac{(\sqrt{U_X^2 + U_Y^2})}{\Delta t}$, where Δt is the time difference between two video frames.

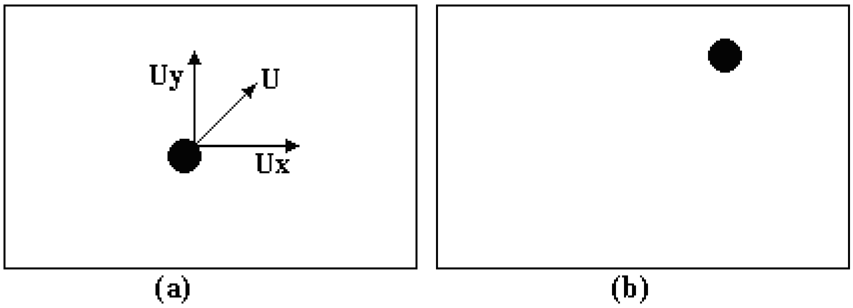


**Figure 1:** Motion vector definition of current frame a) at time t-1 b) at time t .

The DIVA3D Motion Estimation module allows the user to estimate the motion fields between two frames of a given video stream (color or grayscale) and to display the estimated motion field. Because the low level functions used by this module apply only on grayscale (8 bit) video streams, if the input video stream is a color one (24 bits), the block matching function is performed on the video luminance *Y*. The operations of this module are supported by the CameraIO module.

To install the module, the user has to copy the `blockm.dll` in the DIVA3D working directory, where DIVA3D.EXE resides. It will be loaded automatically by DIVA3D ,the next time DIVA3D is started. After installation, the entry *Motion Estimation* is displayed under the *Modules* menu with two sub-menus: *Block Matching* and *Motion Vector Visualization*.

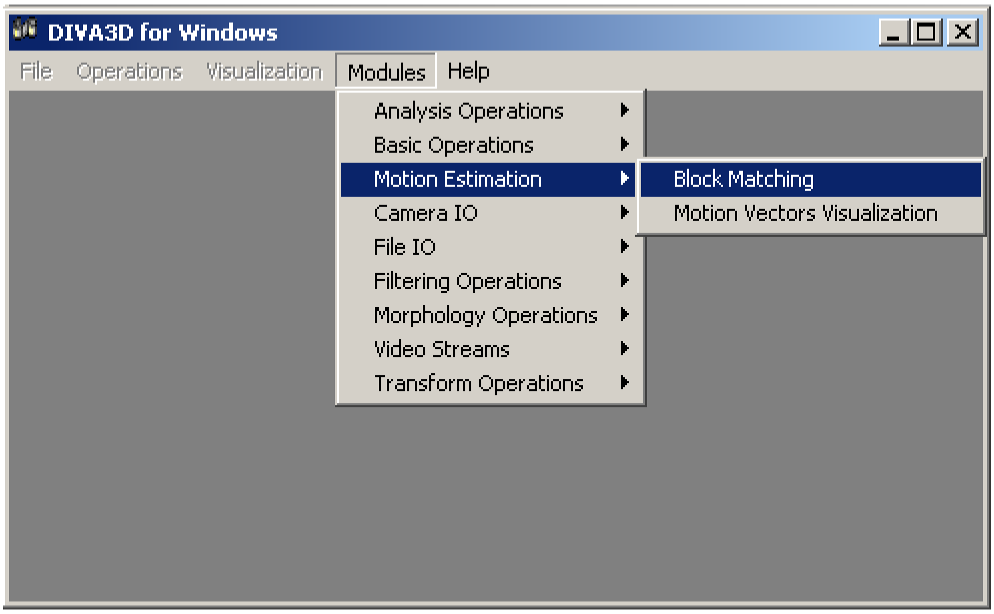


**Figure 2:** The Motion Estimation module submenu.

**Block Matching**

This submenu initializes a block matching technique using the selected motion estimation method in order to perform motion estimation to a pre-selected video stream. A block matching method attempts to find a block of the reference frame (past or future frame) that best matches a predefined block of the current frame. Matching is performed by minimizing a matching criterion which, in this case, is the mean absolute error between a pair of blocks. The block in the reference frame moves inside a search region centered around the position of the block in the current frame. The displacement of the current block with respect to the reference block in $x$ and $y$ directions composes the motion vector assigned to this block.

Five search algorithms have been implemented that aim at reducing computational complexity and increasing motion estimation speed. These are the following ones: full search, the 2D logarithmic search, the three step search, the one at a time conjugate direction search and the orthogonal search. The first method is the slowest, since it performs exhaustive search in the entire search region. Block Matching requests an input video stream from the user, containing the video sequence for which motion estimation will be performed, and generates two integer-valued streams containing the $x$ and $y$ displacements for every frame of the input sequence (measured in pixels), except the first ones. More information on motion estimation can be found in [1] [2].

First the user selects an input video stream and after that the *Block Matching Options* dialog box appears (Figure 3). The fields of this dialog box are:

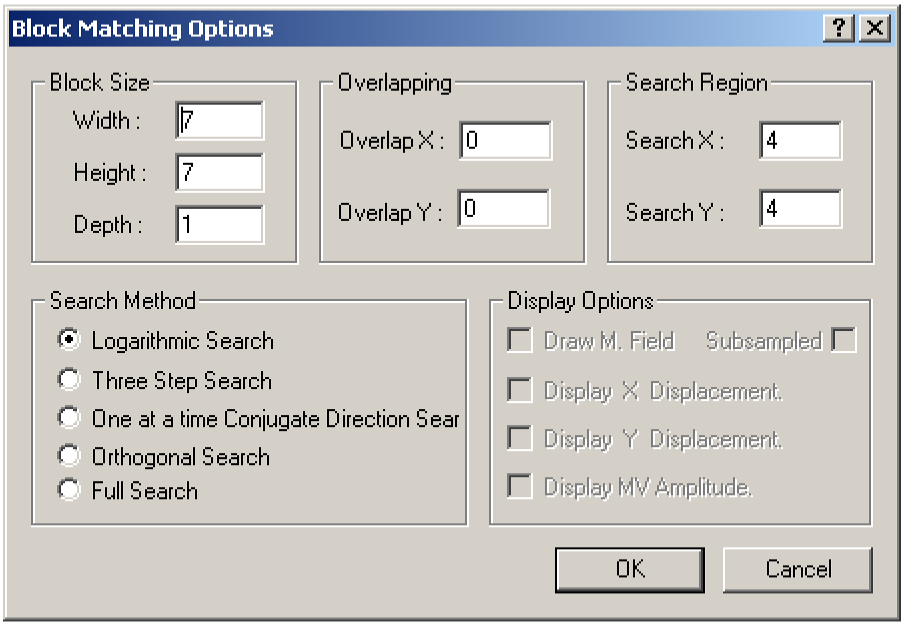


**Figure 3:** Block Matching Options.

- *Block Size:* The fields *Width* and *Height* determine the block *x* and *y* size (in pixels), respectively. The *Depth* field represents the distance in video frames (i.e., in time) that the reference and current frames will have. Its value must be at least 1 and less than the total number of frames of the input video stream. Typically, *Depth* is chosen to be equal to 1, thus allowing motion estimation between two successive video frames.
- *Overlapping:* The fields *Overlap X* and *Overlap Y* determine whether the blocks overlap or not. If their values are both equal to 0, then no overlapping occurs. If they are equal to *Width-1* and *Height-1* respectively, then a motion vector for every pixel of the frame is generated (full overlap). They take values in the range *[0,…..,Width-1], [0,…..,Height-1]* and are measured in pixels.
- *Search Region:* The *Search X* and *Search Y* values assist in the construction of the search region, which is equal to (*Width* + 2·*Search_X) · (Height* + 2·*Search_Y*). The *Search X* and *Search Y* values must be equal, in the case the orthogonal search is selected.
- *Search Method:* It determines which search method will be used. The *Full Search* method performs an exhaustive search for all block shifts inside the search region and selects the location of the minimal matching error. In the *2D Logarithmic Search*, the algorithm follows the direction of minimal matching error, by checking five shifts inside the search region each time and by decreasing the distance between successive the search points [1]. The *Three Step Search* algorithm [1] sets search points at eight, initially coarsely and afterwards finely spaced locations around the centre, following the direction of minimal matching error. The *One-at-a- time Conjugate Search* method [1] is implemented in two steps. In the first

step, the minimal matching error in the horizontal direction is found, whereas, in the second step, the minimal matching error in the vertical direction is searched for. In *Orthogonal Search*, three search points are initially set in the horizontal direction. At the location of the minimal matching error, three search points are selected in the vertical direction this time. The algorithm proceeds this way, until the distance between the search points is reduced to 1.

After setting the options and pressing OK the user is prompted to select two files to save the *ival* (integer-valued) streams containing the output motion vector coordinates and the block matching function is executed. During the block matching process a *Progress* dialog box is displayed (Figure 4).

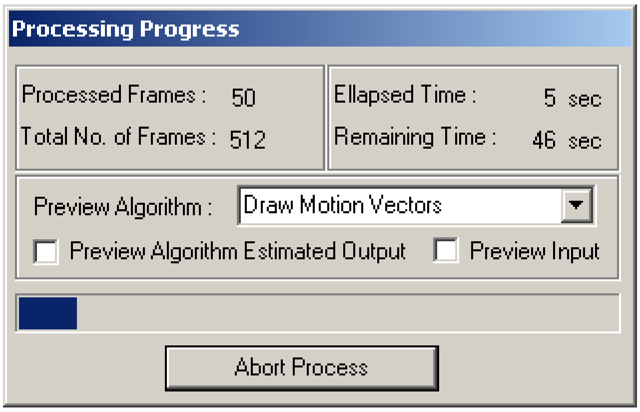


**Figure 4:** Progress dialog box during block matching.

In this dialog box, there are options to select an algorithm to preview the motion vectors during the process and to simultaneously preview input and output.

### Motion Vector Visualization

When selecting to visualize the motion vectors, the user is prompted to select a *ucval* video stream as the source and the corresponding two *ival* (integer-valued) streams containing the *x,y* coordinates of the motion vectors that have been produced by block matching on this video stream. The *Block Matching Options* dialog box is displayed where only the *Display Options* field is activated (Figure 5).

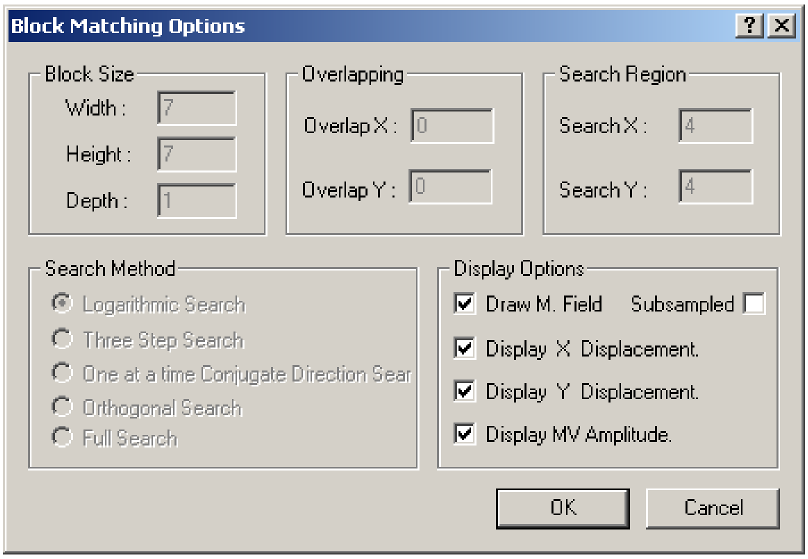


**Figure 5:** Block Matching Options dialog box when Visualizing Motion Vectors

The options in this dialog box are the following:

- *Draw Motion Field:* If this option is selected, the estimated motion field vectors are overlaid on the initial video sequence. The motion vectors start from the corresponding block centre and are directed along motion vector direction.
- *Subsample Motion Field:* Enabling this option leads to displaying a sub-sampled motion field display for better visualization. This option is useful when the motion field is too dense to be displayed in the way described in the *Draw Motion Field* paragraph.
- *Display X Displacement:* This option, when selected, generates a sequence of grayscale images, the gray level values of each one being a scaled version the *x* displacements of the motion vectors. The scaling function translates the range *[smallest_negative_x_displacement, largest_positive_x_displacement]* to the range [0,255] that can be displayed as a grayscale image. A pixel value of 128 represents no *x* displacement for the corresponding block. A value 0 corresponds to the greatest negative *x* displacement and a value 255 to the greatest positive x displacement.
- *Display Y Displacement:* It has the same meaning as the *Display X Displacement*, but the y displacements of the motion vectors are used and displayed.
- *Display MV Amplitude:* The selection of this option displays a sequence of grayscale images, the gray level values of each one being scaled versions of the motion velocity i.e. of the amplitude of the respective motion vectors. A value of 0 translates to no motion, whereas a value of 255 to the largest motion velocity.

Figure 6 shows an example of motion vectors visualization.

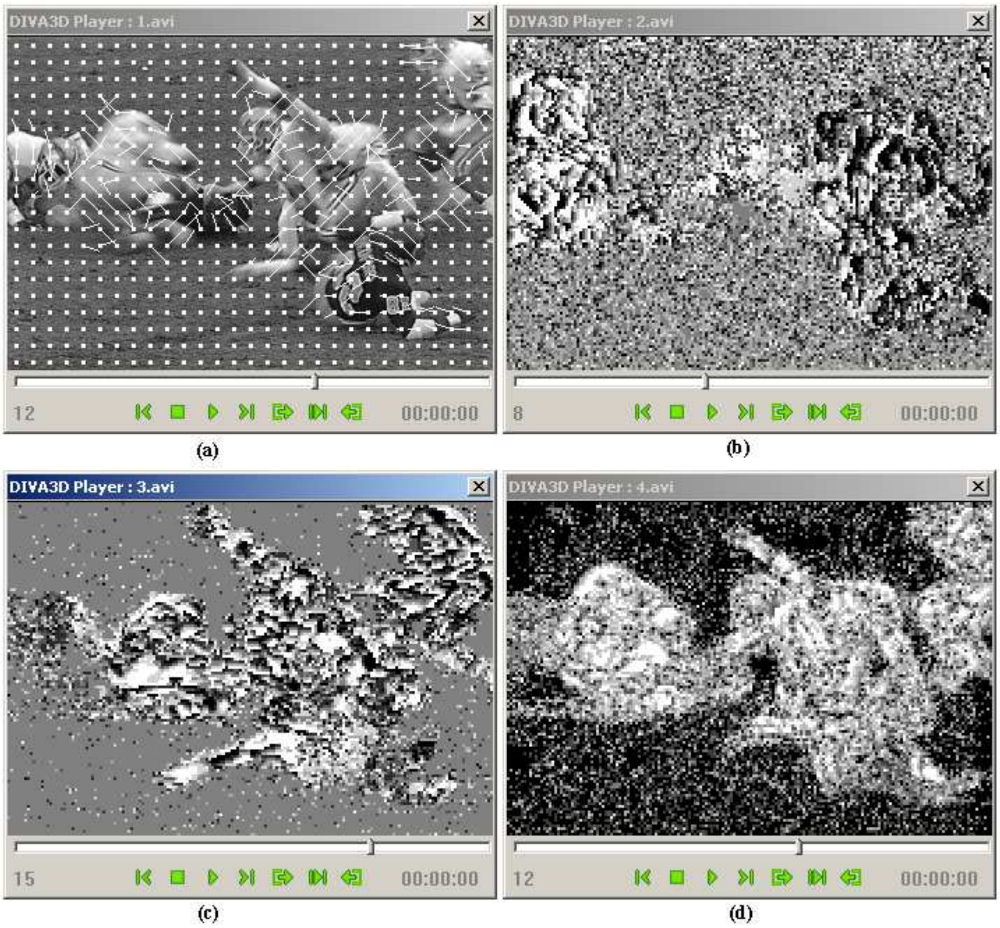


**Figure 6:** An example of displaying the motion field: a) motion field drawing, b) visualization of the *x* coordinate of the motion field, c) visualization of the *y* coordinate of the motion field, d) visualization of the motion velocity.

## Bibliography

[1] A. Murat Tekalp, 'Digital Video Processing', Prenctice Hall 1995
[2] Y. Wang, ‚J.Ostermann,Y.O.Zhang,' Video Processing and Communications', Prentice Hall.